Approximation

GeoComput & ML

27 Apr. 2021

Interpolation

Definition

obtaining some function such that their values are identical to the given data

Definition

for given data

$$(t_i, y_i), \qquad i = 1, \ldots, m$$

we seek a function such that

$$\phi(t_i) = y_i, \qquad i = 1, \ldots, m$$

Motivation

Motivation

finite $\Leftrightarrow$ infinite

discrete $\Leftrightarrow$ continuous

Categorisation

- polynomial
- trigonometric
- piecewise

Polynomial Interpolation

Let $f(x)$ be the unknown function generating the data. We approximate $f(x)$ using a n degree polynomial $\phi_n(x) = \sum_{i=0}^{n} a_i \, x_i^i$ such that

$$f(x_i) = \phi_n(x_i), \qquad i = 0, \ldots, n$$

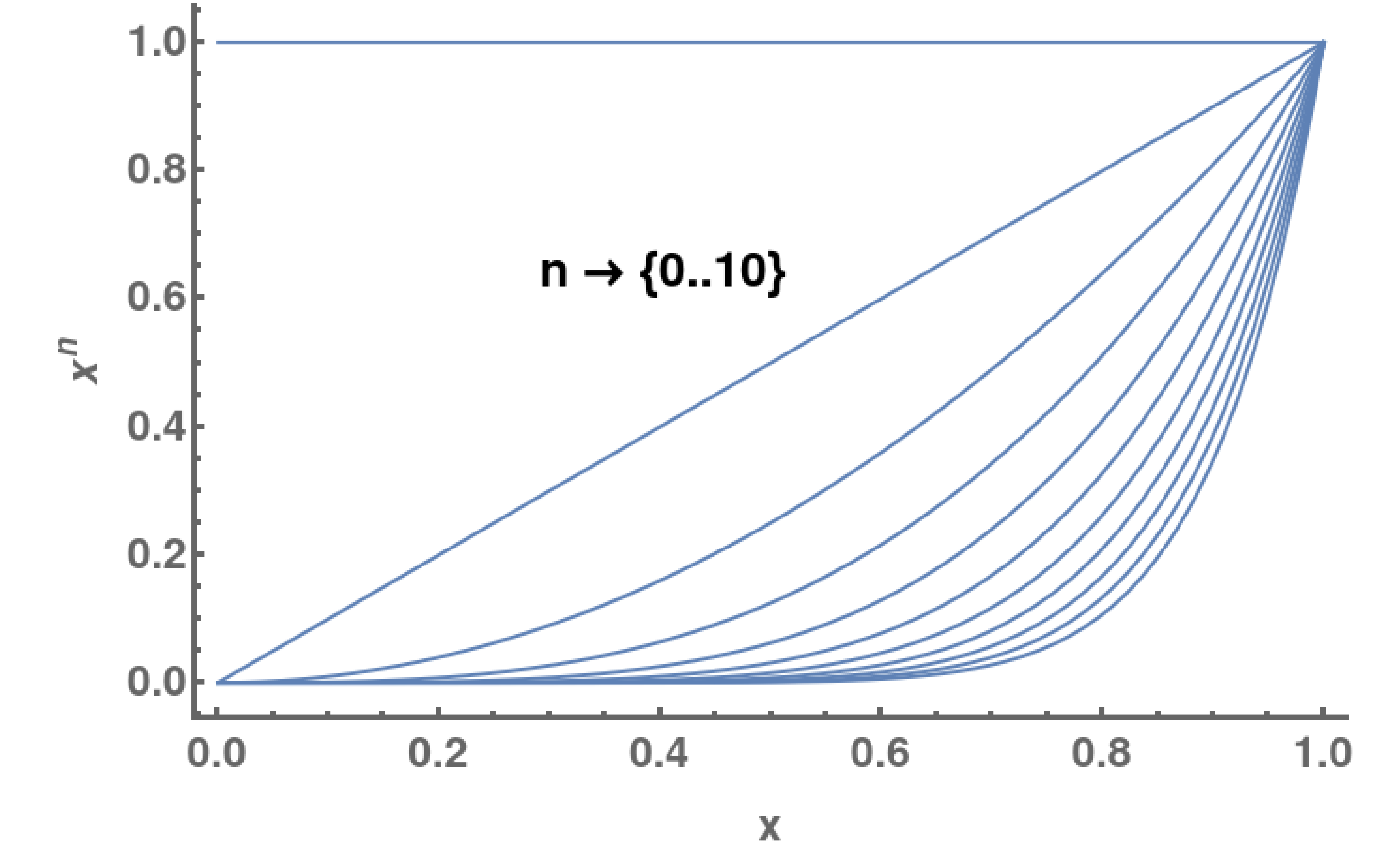
Polynomial Interpolation

that is $f(x_i) = \sum_i^n a_i \, x_i^i$

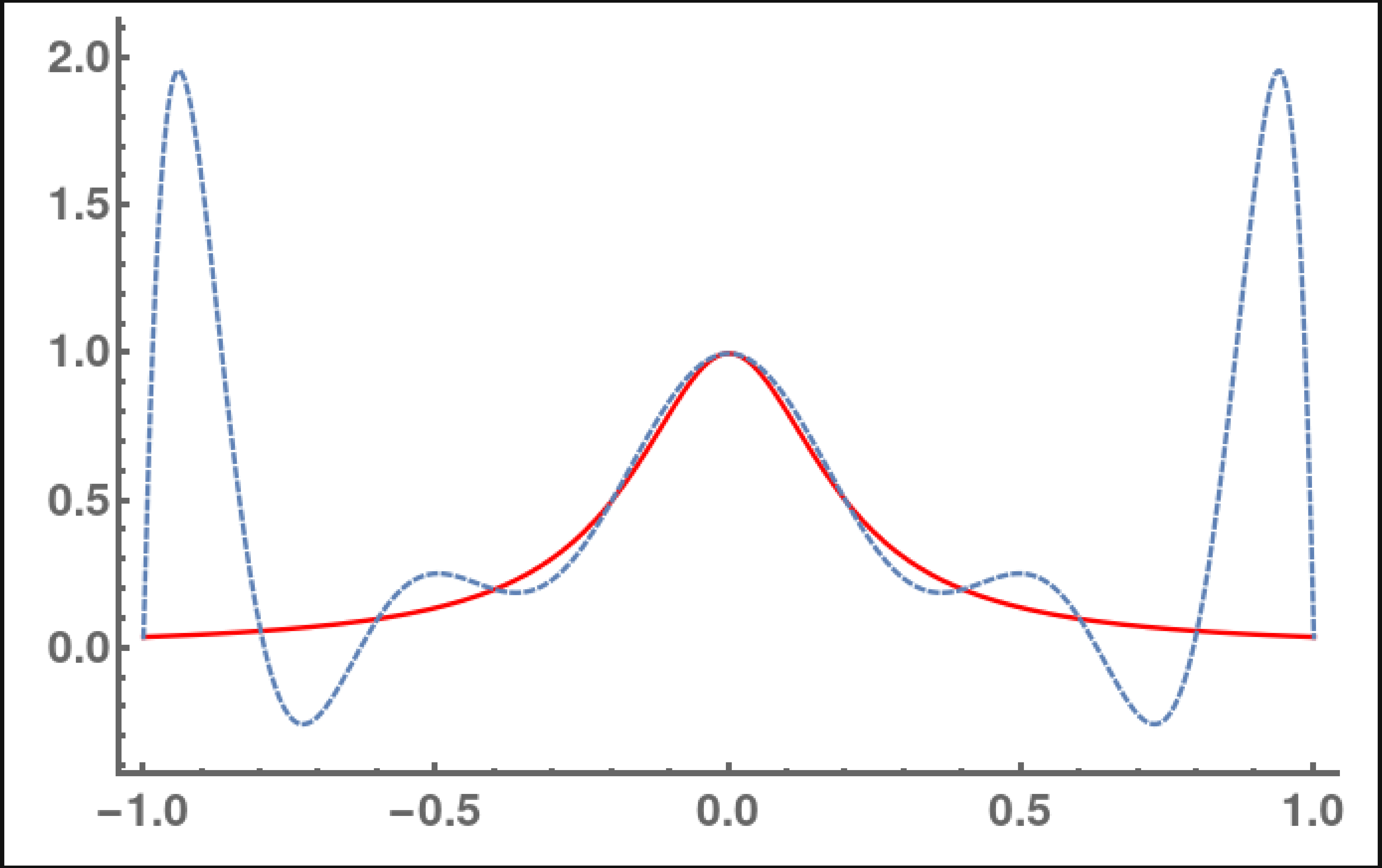
$(n-1)$ linear equations with coefficient determinant

$$\begin{vmatrix} 1 & x_0 & x_0^2 & \dots & x_0^n \\ 1 & x_1 & x_0^2 & \dots & x_1^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \dots & x_n^n \end{vmatrix}$$

Polynomial Interpolation



Polynomial Interpolation



Piecewise Interpolation

Cubic Spline

Generally speaking, a spline is a polynomial of degree k with $k-1$ times continuous differentiabilities.

Cubic Spline

Let $f(x)$ be a function defined in the domain $a \le x \le b$. We partition the function into subintervals $a \le x_0 < x_1 \ldots < x_n \le b$

We aim to find a cubic function $s_{3,i}(x)$ such that

$$s_{3,i}(x_i) = f(x_i), \quad i = 0, \ldots, n-1$$

Cubic Spline

in each subinterval $[x_{i-1}, x_i]$, cubic spline $s_{3,i-1}(x_{i-1})$ must meet :

1. $s_{3,i-1}(x_{i-1}) = f(x_{i-1})$ and $s_{3,i}(x_i) = f(x_i)$
2. $s_{3,i}(x_i) = s_{3,i+1}(x_i)$
3. $s'_{3,i}(x_i) = s'_{3,i+1}(x_i)$
4. $s''_{3,i}(x_i) = s''_{3,i+1}(x_i)$

Cubic Spline

Question :

A cubic spline polynomial has $4(n-1)$ parameters to be determined. How many parameters can be fixed based on the previous constrains?

Hermite cubic spline

Hermite condition

$$H_{3,i-1}(x_{i-1}) = f(x_{i-1}), \quad H_{3,i}(x_i) = f(x_i)$$

$$H'_{3,i-1}(x_{i-1}) = f'(x_{i-1}), \quad H'_{3,i}(x_i) = f'(x_i)$$

Akima

Given a set of knot points (x_i, y_i) with x_i strictly increasing, Akima spline go through all the points and determine the slope for each point as a weighted average of the scants of two points before and after.

$$\boldsymbol{s}_i = \frac{|m_{i+1} - m_i| m_{i-1} + |m_{i-1} - m_{i-2}| m_i}{|m_{i+1} - m_i| + |m_{i-1} - m_{i-2}|}$$

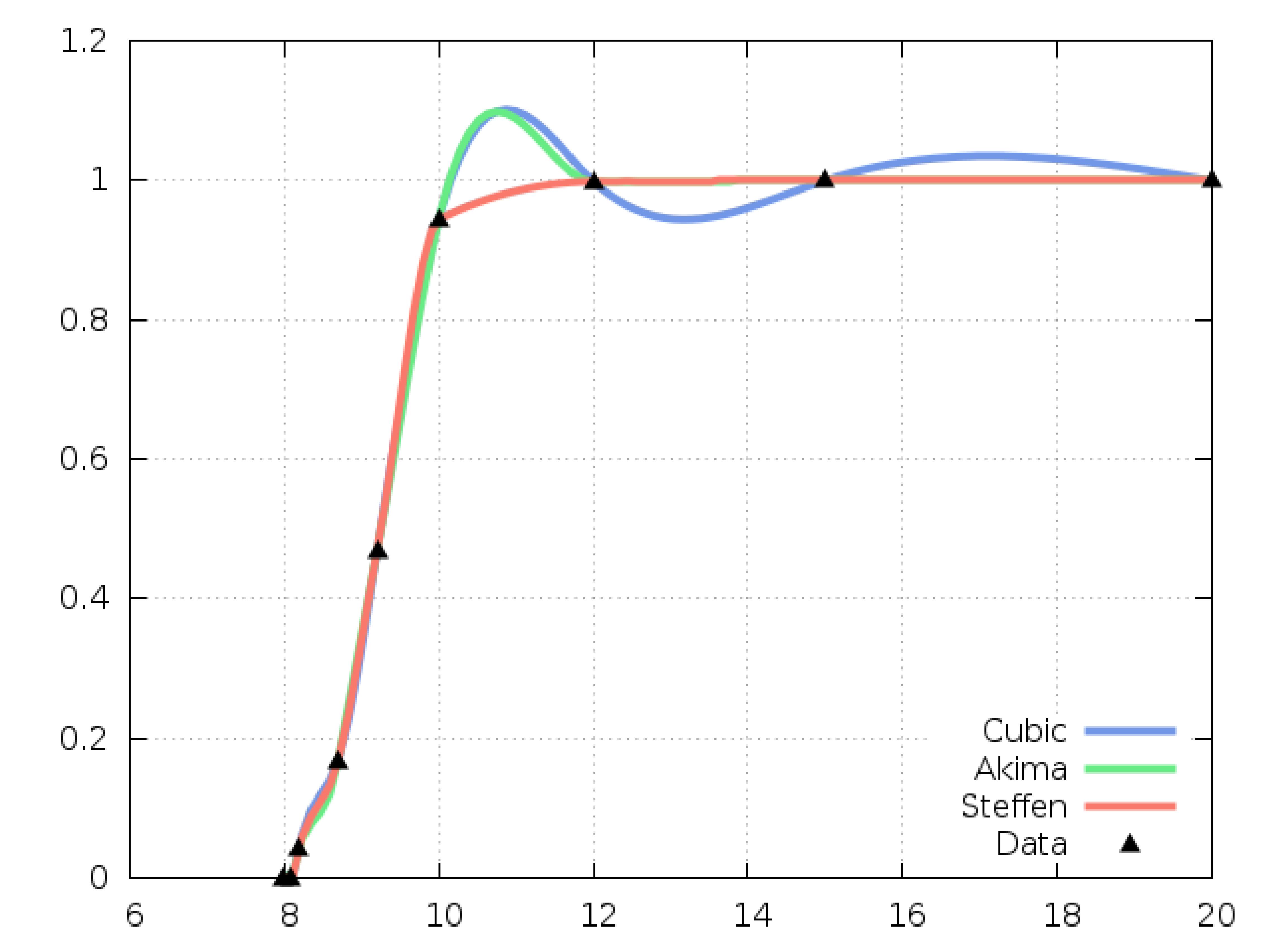
Steffen

estimate the slope of internal points through a unique parabola determined by three neighbouring points to ensure the monotonic behaviour of interpolation

$$p_i = \frac{s_{i-1}h_i + s_i h_{i-1}}{h_{i-1} + h_i}$$

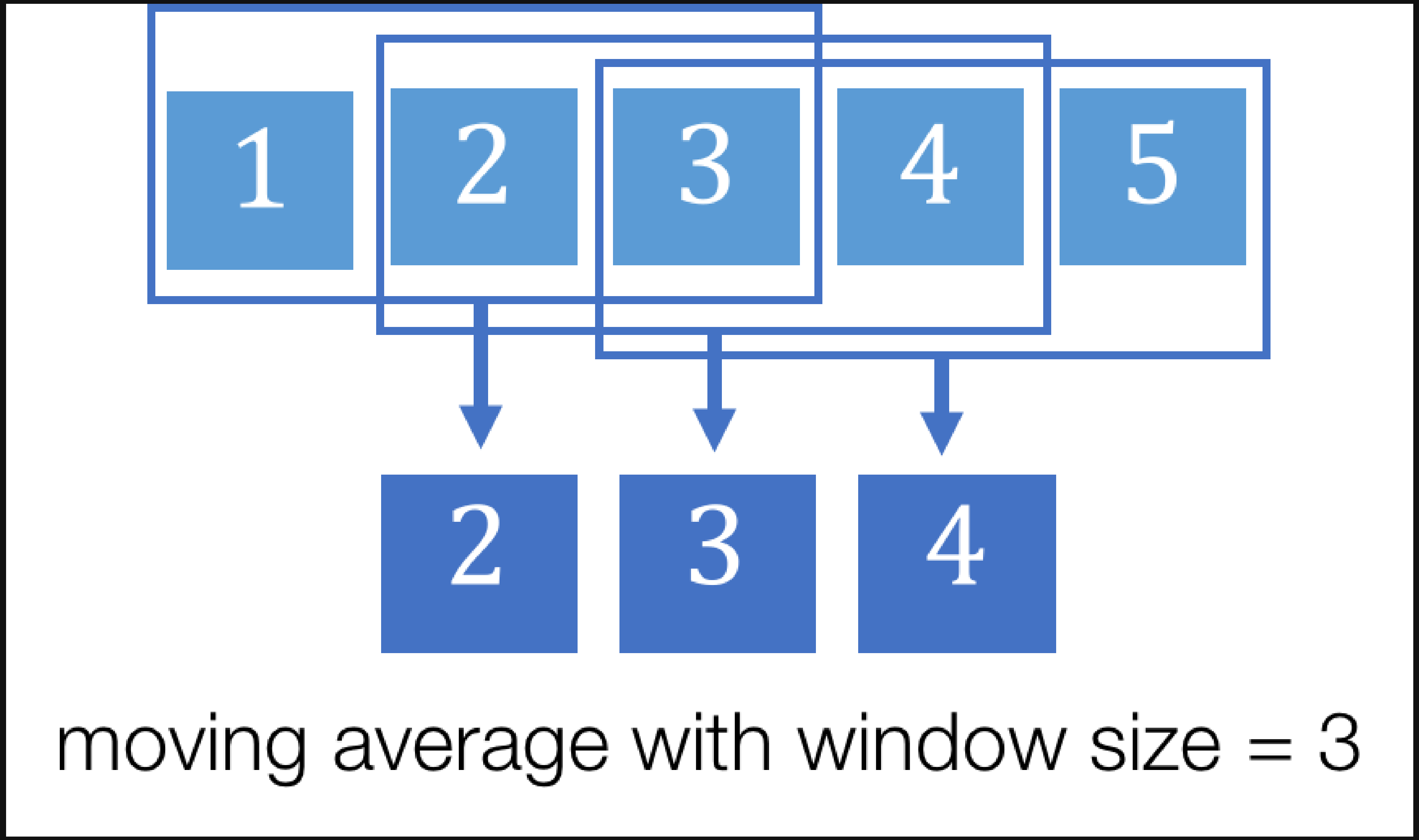
where $h_i = x_{i+1} - x_i$ and $s_i = \dfrac{y_{i+1} - y_i}{x_{i+1} - x_i}$

Comparison



Smoothing

Moving window



$$x_i^* = \frac{1}{2m+1} \sum_{j=-m}^{m} x_{i+j}$$

Salvitsky-Golay filtering

regression fitting

$$x_j^i = \sum_{l=0}^{k-1} a_l j^l, \quad j \in [-m, m], i \in [1, n]$$

$$\boldsymbol{x} = \boldsymbol{M}\,\boldsymbol{a}$$
$$\boldsymbol{a} = (\boldsymbol{M}^T\boldsymbol{M})\boldsymbol{M}\,\boldsymbol{x}$$
$$\hat{\boldsymbol{x}} = \boldsymbol{M}(\boldsymbol{M}^T\boldsymbol{M})\boldsymbol{M}\,\boldsymbol{x}$$

Fourier Transform

Fourier Series

representation of a function $f(x)$ in terms of a set of trigonometric functions

$$cos(n\,x), \quad n = 0, 1, 2, 3, \dots$$
$$sin(n\,x), \quad n = 1, 2, 3, \dots$$

Fourier Series

- orthogonality

$$\int_{-\pi}^{\pi} cos\, mx\ cos\, nx\ dx = 0, \quad m \neq n$$
$$\int_{-\pi}^{\pi} sin\, mx\ sin\, nx\ dx = 0, \quad m \neq n$$
$$\int_{-\pi}^{\pi} sin\, mx\ cos\, nx\ dx = 0, \quad any\ m, n$$

$$\int_{-\pi}^{\pi} cos\, nx\ cos\, nx\ dx = 2\pi\ or\ \pi, \quad if\ n = 0\ or\ n > 0$$
$$\int_{-\pi}^{\pi} sin\, nx\ sin\, nx\ dx = \pi, \quad if\ n = 0$$

Fourier Series

$$\int_{-\pi}^{\pi} cos\,mx\;cos\,nx\;dx = \frac{1}{2}\int_{-\pi}^{\pi}(cos(m+n)x + cos(m-n)x)\,dx$$

$$= \frac{1}{2}\left[\frac{sin(m+n)x}{m+n} + \frac{sin(m-n)x}{m-n}\right]_{-\pi}^{\pi}$$

$$= 0$$

Fourier Series

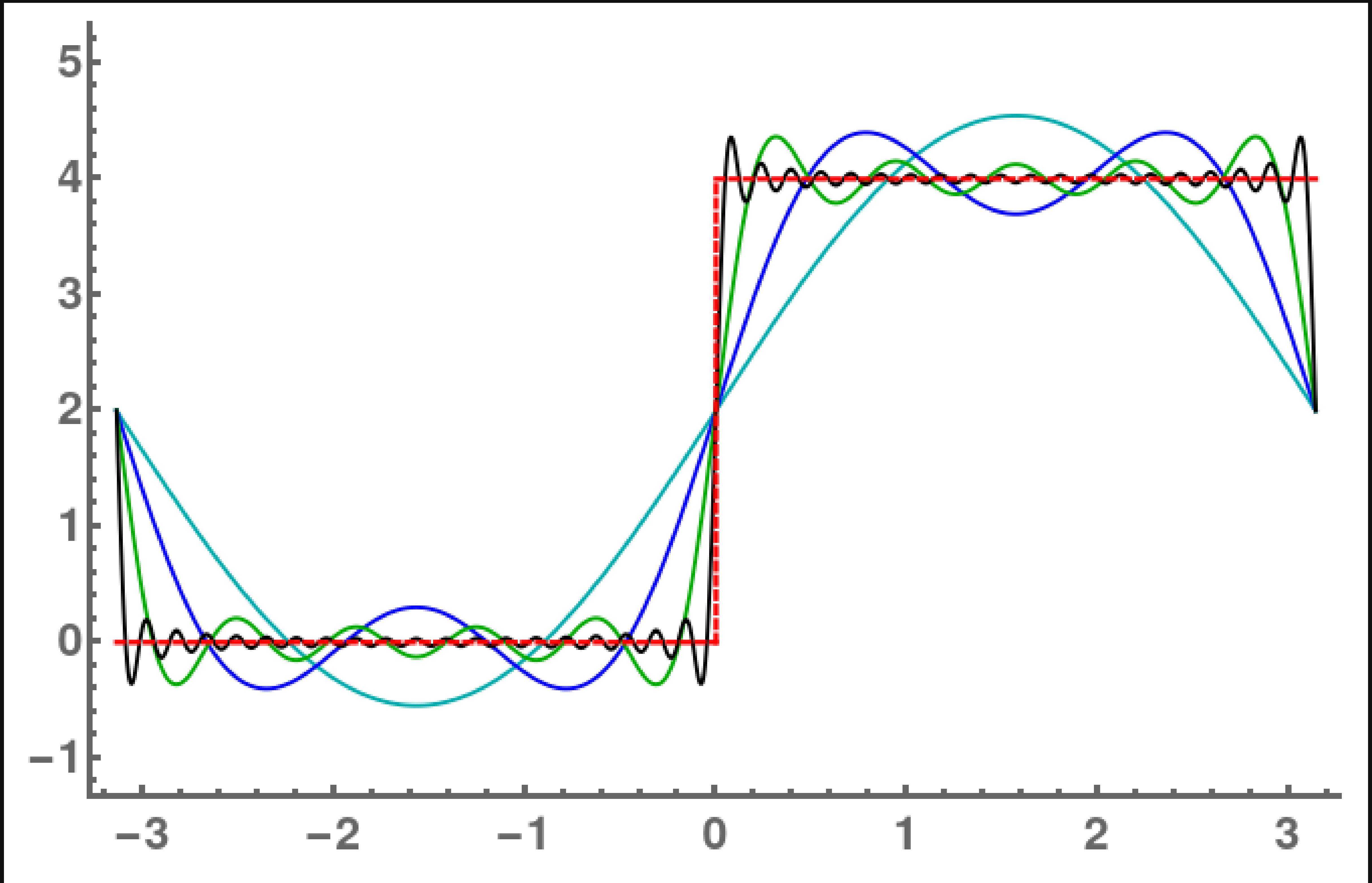
Fourier series

$$f(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty} (a_n cos\, nx + b_n sin\, nx)$$

Fourier coefficients

$$\begin{cases} a_n = \int_{-\pi}^{\pi} f(x) cos(nx)\, dx \\ b_n = \int_{-\pi}^{\pi} f(x) sin(nx)\, dx \end{cases}$$

Fourier Series



Fourier Transform

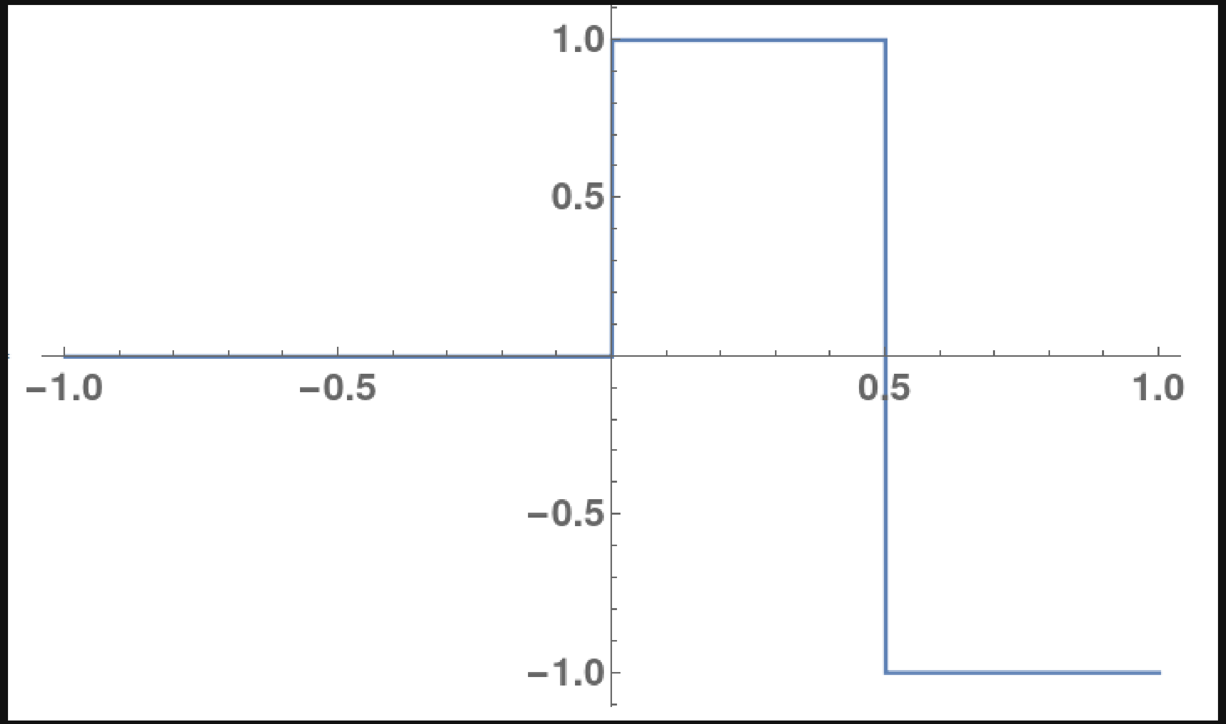
$$F\{f(x)\} = \hat{f}(\omega) = \int_{-\infty}^{\infty} f(x)e^{-i\omega x}dx$$

$$F^{-1}\{\hat{f}(\omega)\} = f(x) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \hat{f}(\omega)e^{i\omega x}d\omega$$

Wavelet

Haar Wavelets

$$\psi(X) = \begin{cases} -1 & 0 \leq x \leq 1/2 \\ 1 & 1/2 \leq x \leq 1 \\ 0 & otherwise \end{cases}$$



Haar Wavelets

for each pair of $j, k \in \mathbb{Z}$

$$\psi_{j,k}(x) = \frac{1}{\sqrt{2^j}} \left(\frac{x - 2^j\, k}{2^j} \right)$$

$$\mathcal{H} = \{\psi_{j,k}(x) \,|\, j, k = \ldots,\ -2,\ -1,\ 0,\ 1,\ 2,\ \ldots\}$$

$$\mathcal{H} = \begin{cases} \dfrac{-1}{\sqrt{2^j}} & [2^j\, k, 2^j\, k + 2^{j-1}) \\ \dfrac{1}{\sqrt{2^j}} & [2^j\, k + 2^{j-1}, 2^j(k+1)) \\ 0 & outside\ [2^j\, k, 2^j(k+1)] \end{cases}$$

Haar Wavelets

$$f(x) = \sum_{j=-\infty}^{+\infty} \sum_{k=-\infty}^{+\infty} d_{j,k} \psi_{j,k}(x)$$

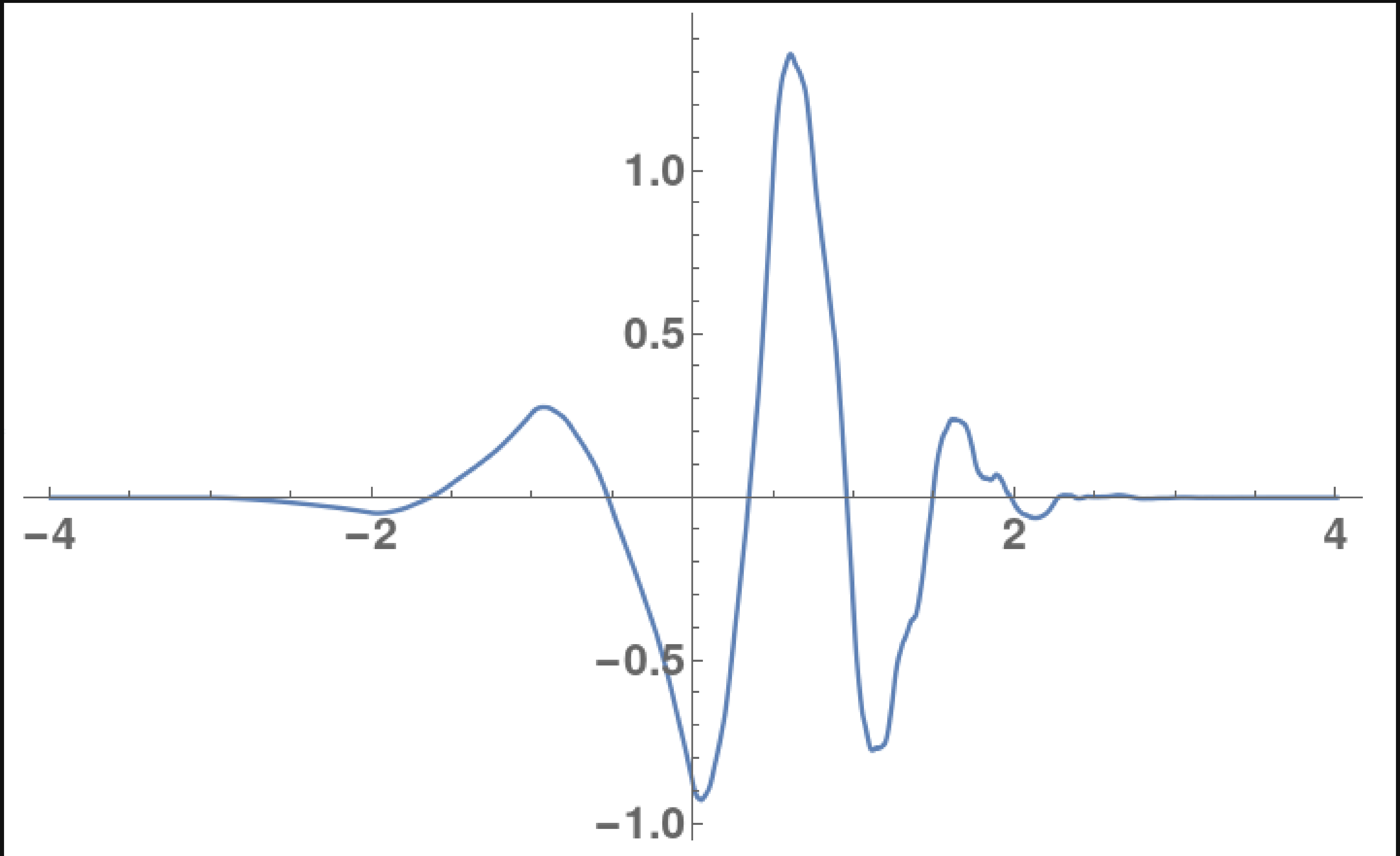
where $d_{j,k} = \langle f(x), \psi_{j,k}(x) \rangle$, wavelet coefficients

Daubechies Wavelets

$$D(\omega) = \sum_{k=0}^{n-1} h_k \, e^{ik\omega}$$

$$\begin{cases} h_0 = \dfrac{1}{4\sqrt{2}}(1+\sqrt{3}) \\ h_1 = \dfrac{1}{4\sqrt{2}}(3+\sqrt{3}) \\ h_2 = \dfrac{1}{4\sqrt{2}}(3-\sqrt{3}) \\ h_3 = \dfrac{1}{4\sqrt{2}}(1-\sqrt{3}) \end{cases}$$

Daubechies Wavelets



Acknowledgement

Thanks for Your Attention

References

- H. Akima, JACM. (1970) 17, 589
- M. Steffen, Astron. Astrophys. (1990) 239, 443
- M. Heath, Scientific Computing An Introductory Survey, 2018
- S. Ray, Numerical analysis with algorithms and programming, 2016
- P. van Fleet, Discrete wavelet transformations _ an elementary approach with applications, 2008
- D. Sundararajan, Discrete Wavelet Transform_ A Signal Processing Approach
- J. Winefordner, Chemometrics From Basics to Wavelets, 2004
- E. Steiner, The Chemistry Maths Book, 1996
- M. Greenberg, Advanced Engineering Mathematics, 2004
- https://www.gnu.org/software/gsl/doc/html/interp.html
- https://dawn.cs.stanford.edu/2017/08/07/asap/